

BAB V

KESIMPULAN

5.1. Pendahuluan

Pada bab ini akan menyatukan hasil temuan dalam penelitian ini. Pada bagian pertama, hasil kesimpulan dari penelitian ini. Pada bagian kedua menggambarkan keterbatasan dari penelitian ini. Bagian ketiga adalah saran untuk penelitian mendatang.

5.2. Kesimpulan

Hasil kesimpulan yang didapatkan dari penelitian yang dilakukan adalah sebagai berikut :

1. Penerapan manajemen risiko dengan menggunakan pendekatan JIT yang diterapkan pada proyek perangkat lunak Kotabagus di PT.Cerise menganalisis risiko di PT.Cerise pada proyek Kotabagus. Pendekatan JIT bertumpu pada elemen risiko *technical*, *cost*, *schedule*. Faktor-faktor risiko mempengaruhi 50% kendala pada risiko *technical* (memicu kegagalan produk perangkat lunak), 48% kendala pada risiko *cost* (memperbanyak biaya), dan 48% faktor *schedule* (memperlama waktu pengerjaan perangkat lunak).
2. Nilai keberhasilan proyek perangkat lunak Kotabagus adalah 0,51, ini menunjukkan bahwa risiko sebesar 49% masih terjadi pada proyek perangkat lunak Kotabagus di PT.Cerise.

3. Penelitian pada proyek perangkat lunak Kotabagus di PT.Cerise Yogyakarta menemukan beberapa faktor risiko yang berisiko tinggi. Faktor risiko yang menjadi prioritas adalah faktor *reliability* menyebabkan risiko sebesar 61% pada proyek perangkat lunak Kotabagus, faktor *risk culture* menyebabkan risiko sebesar 60% pada proyek perangkat lunak Kotabagus, dan faktor *estimation* yang menyebabkan risiko sebesar 52% pada proyek perangkat lunak Kotabagus. Faktor-faktor risiko yang berisiko tinggi atau menjadi prioritas ini menjadi pertimbangan perusahaan dan manajer proyek dalam menyusun strategi rekomendasi untuk pengurangan risiko.
4. Strategi yang direkomendasikan kepada perusahaan dan manajer proyek untuk pengurangan risiko berdasarkan faktor-faktor risiko yang menjadi prioritas adalah menerapkan model keandalan perangkat lunak untuk memprediksi keandalan perangkat lunak. Menerapkan model estimasi untuk memperkirakan biaya, jadwal, dan teknis dan melakukan dokumentasi estimasi penjadwalan, biaya, dan teknis. Melakukan dokumentasi, sosialisasi, dan pelaksanaan prosedur manajemen risiko di perusahaan untuk pengambilan keputusan.

5.3. Saran

Saran yang dapat dilakukan untuk penelitian-penelitian mendatang adalah sebagai berikut:

1. Menambahkan simulasi perhitungan perbaikan untuk risiko-risiko yang menjadi prioritas dan melakukan perbandingan dengan nilai-nilai risiko saat ini.
2. Melakukan optimasi pada salah satu faktor-faktor risiko proyek perangkat lunak pada perusahaan sehingga dapat meningkatkan nilai keberhasilan proyek.



Daftar Pustaka

- Afrizal, Y. & Harjoko, A., 2009. Perangkat Lunak JIT (Just In Time) untuk Memprediksi Resiko Proyek Perangkat Lunak. *Jurnal Sistem Informasi*, 4(1), pp. 61-74.
- Boehm, B. W., 1988. A Spiral Model Software Development and Enhancement. *IEEE*, May, 21(5), pp. 61-72.
- Boehm, B. W., 1991. Software Risk Management : Principles and Practices. *IEEE Software*, 8(1), pp. 32-41.
- Charette, R., 1989. *Software Engineering Risk Analysis and Management*. New York: McGraw-Hill.
- Halloway, C. A., 1979. *Decision Making Under Uncertainty : Models And Choices..* s.l.:Englewood Cliffs: Prentice-Hall.
- Humphrey, W., 1989. *Managing the Software Process*. s.l.:Addison-Wesley/SEI series in Software Engineering Reading.
- Karolak, D. W., 1998. *Software Engineering Risk Management: Finding Your Path Through the Jungle*. s.l.:Prentice-Hall.
- Karolak, D. W., 1999. *Software Engineering Risk Management*. Los Alamitos, CA, USA: IEEE Computer Society Press.
- McCall, J. A., Richards, P. K. & Walters, G. F., 1977. *Factors in Software Quality*, Griffiths Air Force Base, N.Y.: Rome Air Development Center Air Force Systems Command.
- Pressman, R. S., 2001. *Software Engineering : A Practitioner's Approach*. s.l.:Pressman Inc.
- Stern, R. & Arias, J. C., 2011. Review of Risk Management Methods. *Business Intelligence Journal*, 4(1), pp. 59-78.
- Suselo, T., 2007. Analisis Manajemen Risiko Perangkat Lunak dengan Pendekatan Just-in-Time : Studi Kasus Optimasi Organisasi dan Dokumentasi pada Organisasi Pengembang Perangkat Lunak. *Jurnal Teknologi Industri*, XI(2), pp. 13-24.



DAFTAR PERTANYAAN

Organizations					
No	Pertanyaan	0	1	2	3
O1	Apakah perusahaan berencana atau sudah menggunakan manajer proyek perangkat lunak yang berpengalaman?				
O2	Apakah perusahaan telah membuat proyek perangkat lunak yang mirip sebelumnya?				
O3	Apakah perusahaan sudah merencanakan atau mempunyai dokumentasi struktur organisasi?				
O4	Apakah perusahaan mempunyai struktur organisasi tetap?				
O5	Apakah tingkat kepercayaan diri tim proyek tinggi?				
O6	Apakah ada komunikasi yang baik dengan perusahaan yang berbeda dalam mendukung proyek pengembangan perangkat lunak?				
O7	Apakah perusahaan melakukan manajemen konfigurasi perangkat lunak?				
O8	Apakah perusahaan melakukan manajemen kualitas perangkat lunak?				
Estimation					
No	Pertanyaan	0	1	2	3
E1	Apakah proyek pengembangan proyek perangkat lunak di perusahaan menggunakan metode estimasi/perkiraan? (guess, analogy, price to win, dictated by circumstances, top-down, bottom up, other)				
E2	Apakah proyek pengembangan perangkat lunak di perusahaan menggunakan model biaya perangkat lunak?				
E3	Apakah perkiraan/estimasi berdasarkan metrik produktivitas proyek perangkat lunak masa lalu?				
E4	Apakah perkiraan/estimasi dijadwalkan berdasarkan proyek perangkat lunak masa lalu?				
E5	Apakah perkiraan/estimasi direvisi secara bulanan atau lebih sering?				
E6	Apakah perkiraan/estimasi biaya masa lalu anda akurat jika dibandingkan dengan biaya sebenarnya?				
E7	Apakah jadwal perkiraan masa lalu anda akurat jika dibandingkan dengan jadwal sebenarnya?				
Monitoring					
No	Pertanyaan	0	1	2	3
M1	Untuk setiap pengelolaan proyek perangkat lunak, apakah ada pencapaian ideal yang ditetapkan untuk setiap tahap pengembangan proyek perangkat lunak?				
M2	Apakah perincian WBS (<i>Work Breakdown Structure</i>)				

	digunakan untuk rekam jejak dan laporan, biaya dan anggaran untuk setiap bagian pengembangan proyek perangkat lunak?				
M3	Apakah dipersiapkan sistem pemantauan untuk biaya, jadwal, dan lainnya?				
M4	Apakah biaya, jadwal, dan laporan nilai lainnya tersedia sesuai permintaan?				
M5	Apakah biaya, jadwal, laporan lainnya diperbarui setiap bulan/lebih sering?				
M6	Apakah ada masalah log atau tindakan log? digunakan dan diperbarui setiap minggu? (log : daftar rinci informasi perangkat lunak, kinerja sistem atau aktivitas pengguna)				
M7	Apakah terdapat sarana untuk mencatat dan menyimpan masalah teknis? digunakan dan diperbarui setiap minggu?				
Development and Methodology					
No	Pertanyaan	0	1	2	3
DM1	Apakah metodologi pengembangan proyek perangkat lunak didokumentasikan atau direncanakan untuk proyek ini? dan saat ini sudah ditaati dengan ketat?				
DM2	Apakah pengembang proyek perangkat lunak terlatih dalam metodologi pengembangan perangkat lunak?				
DM3	Apakah pelaksanaan metodologi pengembangan perangkat lunak diikuti?				
DM4	Apakah metodologi pengembangan proyek mencakup persyaratan, desain, ulasan kode/penelusuran/pemeriksaan?				
DM5	Apakah metodologi pengembangan proyek membutuhkan rencana uji/prosedur tes untuk semua fungsi perangkat lunak?				
DM6	Apakah metodologi pengembangan proyek membutuhkan dokumentasi seperti persyaratan, desain dan map pengembangan perangkat lunak?				
DM7	Apakah pengujian regresi dilakukan? Pengujian regresi adalah pengujian proyek perangkat lunak yang telah direvisi (apakah fungsi-fungsi sebelumnya dapat berjalan baik atau sebaliknya)				
Tools					
No	Pertanyaan	0	1	2	3
T1	Apakah pengembang proyek perangkat lunak terlatih menggunakan tools/perangkat?				
T2	Apakah menggunakan tools/perangkat otomatis untuk mendesain perangkat lunak?				
T3	Apakah menggunakan tools/perangkat untuk melakukan pengujian?				
T4	Apakah tools/alat digunakan untuk pembentukan prosedur				

	tes?				
T5	Apakah tools/perangkat digunakan untuk pengujian regresi?				
T6	Apakah menggunakan tools/perangkat untuk rekam jejak?				
T7	Apakah menggunakan tools/perangkat untuk rekayasa ulang (mengidentifikasi karakteristik perangkat lunak yang ada berdasarkan kode, struktur, kamus data dll)?				
T8	Apakah compiler/linker,debugger stabil?				
T9	Apakah tools/perangkat tersedia untuk personil pengembangan proyek perangkat lunak bila diperlukan?				
Risk Culture					
No	Pertanyaan	0	1	2	3
RC1	Apakah perusahaan anda bersedia untuk menukar risiko anggaran tambahan untuk keuntungan tambahan?				
RC2	Apakah perusahaan anda bersedia untuk menukar risiko jadwal tambahan untuk keuntungan tambahan?				
RC3	Apakah perusahaan anda bersedia untuk menukar risiko teknis tambahan untuk keuntungan tambahan?				
RC4	Apakah perusahaan anda bersedia untuk menukar risiko anggaran yang lebih sedikit dengan keuntungan yang lebih sedikit?				
RC5	Apakah perusahaan anda bersedia untuk menukar risiko jadwal yang lebih sedikit untuk keuntungan yang lebih sedikit?				
RC6	Apakah perusahaan anda bersedia untuk menukar risiko teknis yang lebih sedikit untuk keuntungan yang lebih sedikit?				
RC7	Apakah perusahaan anda dikelola dengan jelas?				
RC8	Apakah budaya di perusahaan anda konservatif (bersikap mempertahankan keadaan,kebiasaan) dalam pengambilan keputusan?				
RC9	Apakah perusahaan anda melakukan investasi untuk produk atau teknologi baru?				
RC10	Apakah perusahaan anda cenderung untuk membangun atau membeli produk/teknologi?				
RC11	Apakah manajemen risiko dipraktekkan di perusahaan anda?				
Usability					
No	Pertanyaan	0	1	2	3
U1	Apakah ada manual untuk perangkat lunak?				
U2	Apakah ada bantuan fungsi untuk setiap layar input dan output?				
U3	Apakah pengguna terlibat dalam meninjau prototipe atau versi sebelumnya dari perangkat lunak?				
U4	Apakah user interface dirancang standar industri atau yang				

	familiar/mudah bagi pengguna?				
U5	Apakah waktu respon pengguna telah diidentifikasi?				
U6	Apakah desain dievaluasi untuk meminimalkan penekanan tombol dan entri data?				
Correctness					
No	Pertanyaan	0	1	2	3
C1	Apakah semua persyaratan perangkat lunak telah diidentifikasi dan didokumentasikan?				
C2	Apakah persyaratan perangkat lunak terlihat pada desain?				
C3	Apakah persyaratan perangkat lunak terlihat pada kode?				
C4	Apakah persyaratan perangkat lunak terlihat pada prosedur pengujian?				
C5	Apakah ada, atau diharapkan, banyak perubahan yang dibuat dalam persyaratan?				
C6	Apakah desain perangkat lunak terlihat pada kode ?				
C7	Apakah desain perangkat lunak terlihat pada prosedur pengujian?				
C8	Apakah semua tindakan persyaratan akan ditangani dan dilaksanakan?				
C9	Apakah uji fungsional perangkat lunak akan dilakukan?				
Reliability					
No	Pertanyaan	0	1	2	3
R1	Apakah terdapat kondisi penanganan kesalahan dalam desain perangkat lunak dan kode?				
R2	Jika kondisi kesalahan/error terjadi, apakah proses tetap berjalan?				
R3	Apakah toleransi kesalahan ditetapkan untuk input dan output data?				
R4	Apakah masukan diperiksa untuk nilai valid sebelum proses dimulai?				
R5	Apakah kesalahan hardware terdeteksi dan diproses dalam perangkat lunak?				
R6	Apakah penggunaan jenis data global diminimalkan?				
R7	Apakah data cacat dikumpulkan selama integrasi proyek perangkat lunak?				
R8	Apakah data cacat login dan closed-out menjadi hal utama yang dikirim ke klien?				
R9	Apakah model kehandalan proyek perangkat lunak digunakan untuk memprediksi kehandalan proyek perangkat lunak yang dikerjakan?				
R10	Apakah tes dilakukan dengan perencanaan pengujian?				
R11	Apakah pengujian performa akan dilakukan?				
R12	Apakah pengujian dilakukan oleh tim lain terpisah dari tim pengembang perangkat lunak?				

Personnel					
No	Pertanyaan	0	1	2	3
P1	Apakah sumber daya personil tersedia dan diidentifikasi?				
P2	Apakah sumber daya personil berpengalaman digunakan untuk jenis proyek perangkat lunak yang dikembangkan/dibangun?				
P3	Apakah sumber daya personil berpengalaman dalam lingkungan pengembangan proyek perangkat lunak?				
P4	Apakah sumber daya personil berpengalaman dalam impelementasi bahasa pemograman?				
P5	Apakah jumlah personil pengembangan proyek perangkat lunak sudah maksimal?				